

## REMARKS

Claims 1-44 remain pending in this application. In view of the following remarks, reconsideration of the outstanding office action is respectfully requested.

The Office has rejected claims 1-4, 6-8, 10-15, 17-19, 21-26, 28-30, 32-37, 39-41 and 43-44 under 35 U.S.C. § 102(a) as being anticipated by “A Web-Based Nomadic Computing System”, by Kindberg et al. (Kindberg), and claims 5, 9, 16, 20, 27, 31, 38 and 42 under 35 U.S.C. § 103(a) as being unpatentable over Kindberg in view of “The JINI Architecture for Network-Centric Computing”, by Jim Waldo (Waldo).

In particular, the Office asserts that Kindberg discloses a system for enabling one or more arbitrary components to communicate with each other (page 1, Abstract, lines 1-5), the system comprising: a first component associated with one or more universal interfaces (page 6, Place Managers, lines 6-13); and a second component obtaining one of the one or more universal interfaces associated with the first component and automatically invoking the at least one of the universal interfaces to communicate with the first component (page 9, Setting options on the sink, lines 5-18; page 8, lines 10-19). In addition, the Office asserts that Kindberg discloses computer data signal embodied in a carrier wave for enabling one or more arbitrary components to communicate with each other, the signal comprising: a first source code segment having instructions for causing a first component to obtain one of one or more universal interfaces associated with a second component (page 9, Setting options on the sink, lines 5-18; page 8, lines 10-19); and a second source code segment having instructions for causing the first component to automatically invoke at least one of the universal interfaces to communicate with the second component (page 9, lines 1-10; page 6, Place Managers, lines 6-13).

In addition, the Office asserts that Waldo discloses that a second component interacts with an operating system environment the operating system environment having instructions and data for accessing a data object having the one or more universal interfaces (page 78, A simple set of Conventions: lines 1-20), and one or more universal interfaces comprise object-oriented mobile code having instructions for obtaining, interpreting, viewing or modifying data associated with one or more collections of components (page 79: Jini and Java: lines 3-20).

However, as was submitted previously by Applicants in the Amendment of September 30, 2004, the arguments of which are hereby incorporated by reference, Applicants respectfully submit that neither Kindberg nor Waldo, alone or in combination, disclose or suggest “a second component obtaining one of the one or more universal interfaces associated with the first component and automatically invoking the at least one of the universal interfaces to communicate with the first component,’ as recited by claim 1, ‘automatically invoking at least one of the universal interfaces to communicate with the first component’ as recited in claims 12 and 23, or ‘a second source code segment having instructions for causing the first component to automatically invoke at least one of the universal interfaces to communicate with the second component’ as recited in claim 34.

Instead, Kindberg relates to a well-known “push” function wherein a source component, (i.e. a camera) opens a connection with a sink component (i.e. a printer), and writes content to the sink component. (p. 8, “Direct Content Post,” lines 11-12). In addition, while Kindberg teaches that there may be a simple “content-negotiation,” such as an initial exchange of information between the components, using XML, or another universal language, to identify a file format compatible with both components, there is no disclosure that the initial exchange of information includes a data object that includes instructions and data for invoking one or more universal interfaces. (p. 8, “Direct Content Post,” lines 15-20). Kindberg further discloses that the system needs “some way for the camera to set the configuration of the printer in a device-independent manner.” To do so, Kindberg discloses the use of an XML or HTML form.

Based on the disclosures of Kindberg, Applicants believe that the Office is mistakenly considering a universal interface to be equivalent to a URL. This is incorrect, as will be readily appreciated by a person of ordinary skill in the art. The automatic discovery of URLs from physical surroundings, and using localized web servers for directories, to create location-aware but ubiquitous systems as is disclosed by Kindberg, is not comparable to, much less remotely equivalent to, enabling communication between one or more arbitrary components using a universal interface, as is recited in the claims. After reading the specification and the currently pending claims in the present application, one of ordinary skill in the art at the time of the invention would not confuse the automatic discovery of URLs for accessing web resources that are associated with a physical location disclosed by Kindberg

with the invention recited in the currently pending claims. Also, the Office seems to have equated the first component recited in the claims with a location, and thus further misapplied Kindberg for its disclosures regarding a place manager.

Waldo discloses mobile code and the ability to provide domain-specific interfaces between components. In this regard, it is important to note that Waldo relates only to interfaces that are domain specific. According to Waldo, the first component needs to know in advance that the second component is a fileserver, and then request the file server driver from the second component (see [0004]). This is not equivalent to claimed invention, wherein there is now requirement that the first component know what the second component's domain is prior to the receipt of the universal interface. Thus, according to the present invention, the components only need implement a very limited number of interfaces (just the "bootstrap" interfaces), to automatically obtain the complete interfaces for accessing the respective components.

In addition, contrary to the disclosures of Kindberg and Waldo, the claims of the present application teach that a second component can obtain a universal interface from a first component and automatically invoke that universal interface to communicate with the first component. Each component receives (over a common interface) the device dependent code of the other to allow each component to communicate to the other component using the format and protocols desired by the other component.

The universal interface of the present invention allows components to communicate even when the components do not agree on particular protocols or media encodings. Instead, the components need only agree on a universal interface and the discovery and bootstrapping protocol. This is distinct from Kindberg as Kindberg only allows components to interoperate if they agree on all details of communication, including protocols and encodings. The URL of Kindberg simply allows one component to locate and identify another component—it does not support other dimensions of communication.

As is described in the specification in paragraphs [0033-0034], the claimed invention allows arbitrary components to communicate with each other. According to the present application, universal interfaces includes instructions, sets of operations and/or other data that can be understood and performed by *each* of the components to enable them to communicate

with each other without regard to the domain or to a domain-specific interface. This is distinct from Waldo in that Waldo requires that at least one of the components have knowledge of the domain-specific interface used by the other component. If this condition is not satisfied prior to the discovery process, the components would not be able to communicate.

The claimed invention has no such limitation. Instead, the claimed invention uses universal interfaces to “bootstrap” domain specific operations between devices. For example, if the domain specific operations are file storage and access operations, and a first component wants to access the file storage on a second component, the first component’s discovery mechanism locates the second component and obtains, from the second component, the code and protocols required for the first component to access the file storage of the second component using whatever protocol is desired by the second component (the second component provided the code to the first component such that the first component could then access and utilize the second component). Thus, the first component does not need prior knowledge or awareness of the second component’s protocols but instead the components only need to implement the discovery protocol and the facilities required to request, deliver, and receive the interface that enables the components to operate together. (Para. [0028], [0040], and [0051]).

Accordingly, based on the arguments presented above, Applicants submit that none of claims 1-44 are anticipated under 35 U.S.C. § 102(a) or rendered obvious under 35 U.S.C. § 103(a) in view of Kindberg and Waldo, alone or in combination, and respectfully request that these rejections be withdrawn.

In view of all of the foregoing, Applicants submit that this case is in condition for allowance and such allowance is earnestly solicited.

Respectfully submitted,



Marc S. Kaufman  
Registration No. 35,212

Date: August 17, 2005

NIXON PEABODY LLP  
c/o Gunnar G. Leinberg, Reg. No. 35,584  
Clinton Square, P.O. Box 31051  
Rochester, New York 14603-1051  
Telephone: (585) 263-1014  
Facsimile: (585) 263-1600